Graph-Induced Transformers for Efficient Multi-Hop Question Answering

IR&NLP Lab. Giwon Hong, Jeonghwan Kim, Junmo Kang, Sung-Hyon Myaeng

The 2022 Conference on Empirical Methods in Natural Language Processing (EMNLP 2022)

Motivation

Question

The Schumann–Runge bands are named for at least one German what?

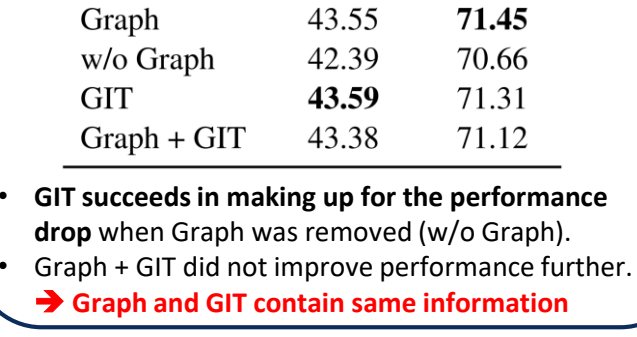
Paragraph A

The Schumann–Runge bands are a set of absorption bands of … 176 and 192.6 nanometres. The bands are named for Victor Schumann and **Carl Runge**.

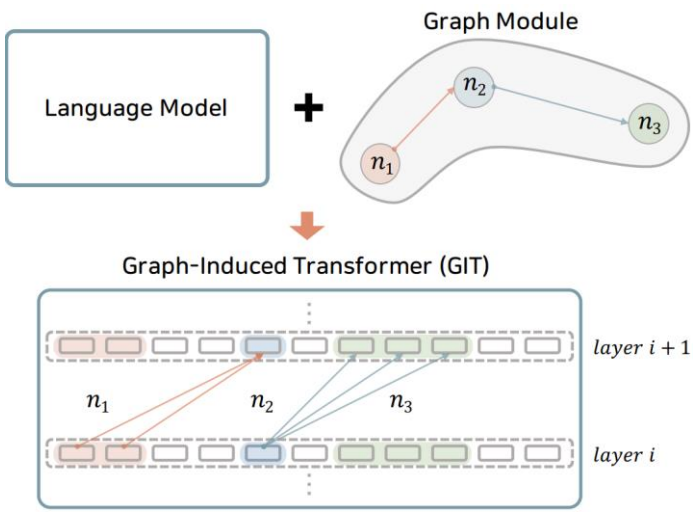
Paragraph B

Carl David Tolmé Runge (30 August 1856 – 3 January 1927) was a German mathematician, physicist, and spectroscopist.

- In **Multi-Hop Question Answering** like HotpotQA[1], connectivity between texts should be exploited.
- Previous works place a graph module on top of a language model to utilize connectivity (SAE[2])

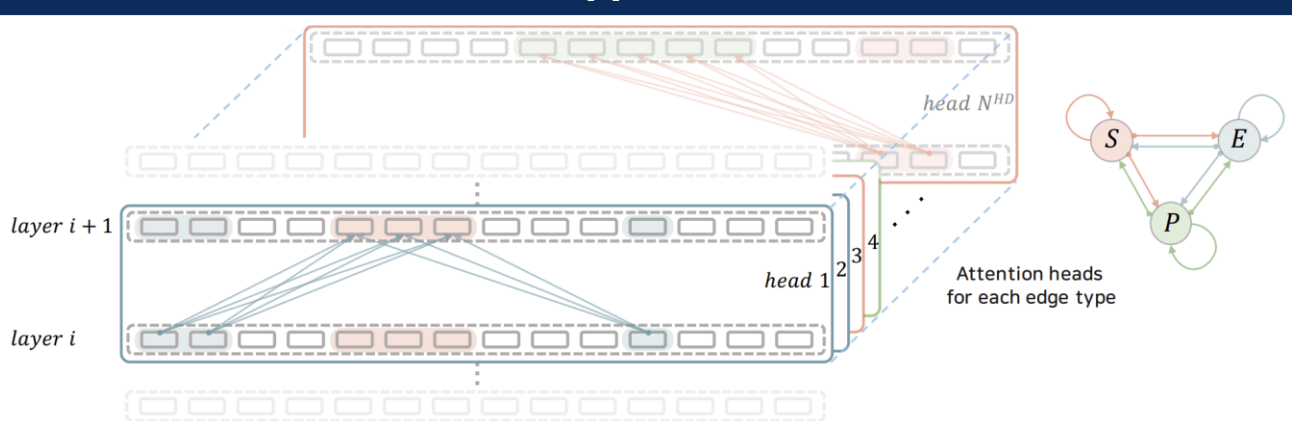


- Unlike pre-trained LMs, these graph modules usually have to be retrained from scratch.
- This results in **sample inefficiency**, i.e., numerous samples required for the necessary inductive bias.



- **Graph-Induced Transformers (GIT)** embeds text graphs inside Transformers.
- Models the text and its structure without additional graph modules ➔ **Sample efficiency.**

Approach

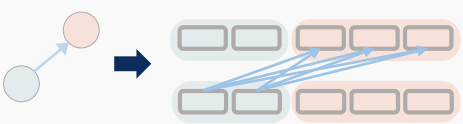


Nodes in GIT

- Nodes in text graphs could be paragraphs, sentences, or entities.
- **GIT represents each node as a set of tokens in the Transformer**

Edges in GIT

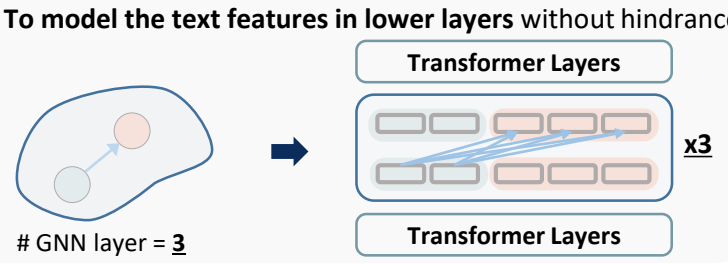
- **GIT represents edges as fully connected attentions between node sets**



GIT Layers

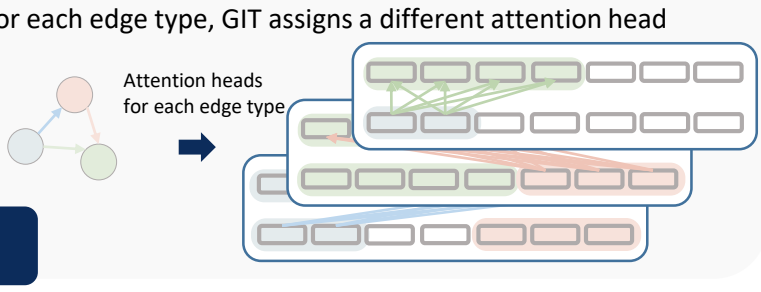
- GIT is selectively applied only to some Transformer layers.

1. **To simulate the number of propagation** of graph modules
2. **To model the text features in lower layers** without hindrance.



Heterogeneous Edge Types in GIT

- For each edge type, GIT assigns a different attention head



Experimental Results

SAE	Joint EM	Joint F1
Graph	43.55	**71.45**
w/o Graph	42.39	70.66
GIT	**43.59**	71.31
Graph + GIT	43.38	71.12

- **GIT succeeds in making up for the performance drop** when Graph was removed (w/o Graph).
- Graph + GIT did not improve performance further.
  ➔ **Graph and GIT contain same information**

Data Portion	SAE	GIT
1%	9.57	**15.68** (↑ 63.8%)
2%	17.79	**24.88** (↑ 39.8%)
5%	28.05	**29.68** (↑ 5.8%)
10%	31.41	**33.02** (↑ 5.1%)
50%	40.61	**41.67** (↑ 2.6%)

- Since the SAE model has to learn the Graph module from scratch, **its performance drops significantly in data-poor environments.**
- GIT utilizes pretrained LM as it is, resulting in sample efficiency.

GIT	Joint EM	Joint F1
Layer 1-24	42.75	70.70
Layer 1-3	42.47	70.72
Layer 8-10	43.07	71.21
Layer 15-17	43.04	70.76
Layer 22-24	43.36	71.01
Layer 21-23	**43.59**	**71.31**
GIT Additional Layers		
Layer 25-27 (No GIT)	43.13	71.07
Layer 25-27	**44.09**	**71.64**

- Applying GIT to the **low-middle layers of the Transformer adversely affects** the performance
- Layer addition improves the performance, but could lead to decreased sample efficiency
- Better not to perturb the task-specific last layer

Acknowledgement

This work was supported by Institute for Information & communications Technology Planning & Evaluation(IITP)

References

[1] Yang et al., 2018; *HotpotQA: A dataset for diverse, explainable multi-hop question answering*, EMNLP 2018

[2] Tu et al., 2020; *Select, Answer and Explain: Interpretable Multi-hop Reading Comprehension over Multiple Documents*, AAAI 2020